

### 1、变量

为什么需要变量？

比如登录：一开始是没有账号和密码这个数据的；

账号和密码来自用户输入；我们怎么去表示账号和密码？

这个时候就需要变量来暂时代替数据，代入到业务实现的逻辑里面；

变量就是用来缓存数据的；

**本质：变量是用来存放数据的空间**

### 2、变量的认知：

- 是弱类型的：我们在声明变量的时候，是不用指定变量类型的；不提前声明变量的类型，在程序运行过程中，类型会被自动确定。
- 变量的声明：使用var(暂时使用var variable)来声明；
- 声明变量特殊情况：

特殊情况	原因	结果
var age; console.log(age);	只声明，不赋值	undefined
console.log(age)	不声明 不赋值 直接使用	报错
age = 10;console.log(age);	不声明 只赋值	10

- 变量的命名：字母（大小写）、数字、下划线、\$，不能用数字开头；
- 严格区分大小写；var age; 和 var Age; 是两个变量
- 变量命名【务必】做到【见名知意】；
- 不要使用保留字（关键词：for if var function class import）；
- 引用没有声明的变量会报错；
- 使用=（一个等号）给变量赋值；==判断是否相等的；=== 全等；!=不全等
- 支持一次性声明多个变量：多个变量之间使用逗号(,)分开；
- 和变量对应的是字面量，也叫直接量；
- 遵循驼峰命名法，首字母小写，后面单词首字母大写 myFirstAge
- 变量类型：使用typeof获取一个变量的类型；变量类型是根据=右边的数据类型来判断的

### 3、基础类型：

#### number

- JS中数值的最大值：Number.MAX\_VALUE
- JS中数值的最小值：Number.MIN\_VALUE

```
1 console.log(Number.MAX_VALUE); //JS中数值的最大值
2 console.log(Number.MIN_VALUE); //JS中数值的最小值
```

不分整数、小数、双精度等，就只有number;

JS不适合进行科学计算（精度要求比交高的）；

支持8进制和16进制；JS中八进制前面加0，十六进制前面加 0x

无穷大 Infinity，无穷小 -Infinity；NaN 代表非数值

```
1 var num1 = 0.3,
2     num2 = 0.2,
3     num3 = 0.1;
4 console.log(num1 - num2);
5 console.log(num2 - num3);
6 //小数精度不够的解决办法,可以把小数转换成整数
7 console.log(num1 * 1000 - num2 * 1000) / 1000;
8
9 //8进制number表示:用0开头，从右向左进行类型，从而转换成10进制
10 //1*8的0次方 + 2 * 8的1次方 + 1 * 8的2次方 = 8*/
11 var num8 = 0121;
12 console.log(num8);
13
14 //16进制
15 var num16 = 0x100;
16 console.log(num16);
17 //无穷大 无穷小
18 console.log('无穷大的保留字', Infinity);
19 console.log('无穷小的保留字', -Infinity);
20
21 console.log(1 + true); //2
```

## string

用引号引起来的；'age', "school",

单引号、双引号、反引号(`，键在数字1的左边)；

使用+进行字符串连接，或者说遇到加号的时候，优先考虑字符串；

length获取字符串的长度;

### 字符串引号嵌套:

JS可以用 **单引号嵌套双引号**, 或者用 **双引号嵌套单引号 (外双内单, 外单内双)**

```
1 var strMsg = '我是一个“大学生”' //可以用 ' ' 包含 " "  
2 var strMsg2 = "我是'大学生'" //可以用" " 包含 ' '
```

```
1 var space = ' '; //注意: 这里是空格, 不是空字符  
2 console.log(space);  
3 var strbnull = ''; //注意: 这里是空字符, 不是空格  
4 console.log(typeof strbnull);
```

### 字符串转义符:

转义符都是 \ 开头的, 常用的转义符及其说明如下:

转义符	解释说明
\n	换行符, n是newline
\\	斜杠\
\'	' 单引号
\"	" 双引号
\t	tab 缩进
\b	空格, b是blank的意思

### 字符串长度

字符串是由若干字符组成的, 这些字符的数量就是字符串的长度。通过字符串的 length 属性可以获取整个字符串的长度。

//通过字符串的length属性可以获取整个字符串的长度

```
1 /* 获取字符串长度 */  
2 var username = '小光',  
3 age = '18';  
4 console.log(username.length, age.length); // 3 2
```

## 字符串的拼接:

// 用 + 进行字符串连接, 并优先考虑字符串运算

```
1 var userName = '小光',
2   age = 21,
3   school = '成都理工大学',
4   major = '电子商务';
5 console.log('我是' + userName + ', 今年' + age + '岁了。' + '在' + school + '上' + major +
6   '专业');
6 // 我是小光, 今年21岁了。在成都理工大学上电子商务专业
```

- 多个字符串之间可以使用 + 进行拼接, 其拼接方式为 字符串 + 任何类型 = 拼接之后的新字符串
- 拼接前会把与字符串相加的任何类型转成字符串, 再拼接成一个新的字符串
- 注意: 字符串 + 任何类型 = 拼接之后的新字符串

## 反引号:

```
1 反引号是字符串模板, 非常常用, 主要用于解决字符串连接
2 var username = '陈100',
3   age = 18,
4   school = '成都理工大学',
5   $major = '电子商务';
6 var description = `我是${username}, 今年${age}岁了,
7   来自${school}${$major}专业`;
8 //${}
```

## boolean

非黑即白 非真既假 非错即对 真真假假 Boolean

true, false 等价于1和0

console.log(1+true) // 2

```
1 console.log(1 + true); //2
```

true 真

false 假

## null

只有一个值null (一个声明变量给 null 值, 里面存的值为空)

类型是object

JS里面一切皆为对象; 万物皆空;

```
1 var n = null;
2 console.log(typeof n, n); //object null
```

## Undefined:

- 只有一个值undefined; 类型是undefined; 表示未定义
- 声明的变量没有赋值, 那么他的值是undefined;
- undefined 和字符串相加, 会拼接成字符串
- undefined 和数字相加, 最后结果是NaN

```
1 var u;
2 console.log(typeof u, u);
3 console.log(undefined == null); // true
4 console.log(undefined === null); // === 考虑变量类型 false
```